VZ 200-300

HUNTER VALLEY

VZ JOURNAL

HUNTER VALLEY VZ USERS' GROUP

THE SECRETARY/EDITOR
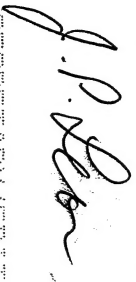JOSEPH P LEON
35 TIGHES TERRACE
TIGHES HILL NSW 2297

(049) 69 2599

25.3.93

DEAR MEMBER,

According to our records your subscription runs out with enclosed issue. If re-subscribing please make cheque/money order/etc payable to: Joseph P LEON.

Yours sincerely

J.P. Leon

SECRETARY/EDITOR
JOSEPH P LEON

JL

FRONT COVER
SHOULD THE JOURNAL CATER FOR IBM TYPE COMPUTERS LIKE SOME MEMBERS HAVE SUGGESTED? I'M INTERESTED IN YOUR OPINION. IN THE MEANTIME READ MY COMMENTS ON PAGE 3 ON THE SUBJECT, ED.

HELP - SELL & TELL   Page 3
CONDOLENCES, FAREWELL HARRY, IS THE END OF THE VZ NEAR?, WANTED TO BUY AND SYSTEM FOR SALE.

VZ BUS MOUSE PART 3 BY L MILBURN   Pages 4-10
THIS IS THE FINAL PART ON THE BUS MOUSE SERIES. ANYONE WANTING THE MOUSE DRIVER AND SOURCE CODE PLEASE CONTACT THE EDITOR.

CHECKDISK2 & PARK2   Pages 11-14
BY DAVE MITCHELL
AFTER A LONG ABSENCE WE WELCOME DAVE BACK WITH A COUPLE OF UPDATED UTILITIES. CHKDSK2 HAS A RE-ENTER OPTION AMONG OTHER IMPROVEMENTS WHILE PARK2 IS FOR 2 DRIVE USE AND CAN ALSO BE USED ON SINGLE DRIVES.

CONVERTING SOURCE CODE UPDATE   Page 14
DAVE MITCHELL HAS BEEN BUSY, HIS EDITOR ASSEMBLER WILL LOAD ALL FOUR SOURCE CODE TYPES WITH NO MODIFICATION AT ALL.

EXT12.2 UPDATE   Page 14
AT MY SUGGESTION LESLIE MILBURN HAS ADDED A COUPLE COMMANDS TO EXT12.1, HENCE THE NAME CHANGE.

INTRODUCTION TO PROGRAMMING   Pages 15-17
BY BOB KITCH
AFTER SOME REQUESTS BOB WILL TRY AND HELP AND EXPLAIN WHAT PROGRAMMING IS ALL ABOUT AND HOW TO DO IT PROPERLY.

UNDERSTANDING COMPUTER TECHNOLOGY   Page 17
THE BEST EXPLANATION OF COMPUTER TECHNOLOGY I HAVE SEEN IN A LONG TIME.

TECHNICAL DATA SHEETS # 5 & 6   Pages 18-19
TWO VZ200 PRINTER INTERFACES FOR THOSE WITHOUT A VZ TECHNICAL REFERENCE MANUAL AND DECODING Z80 64K MEMORY.

DAVE MITCHELL SOFTWARE FOR SALE   Page 20
PATCH3.3 - EXT DOS
MENU/FILE COPIER

USER GROUPS * NEWS * SUBSCRIPTIONS   Page 20

BELIEVE IT OR NOT
WE ALL HEARD STORIES ABOUT THE POSTAL SERVICE AND HOW LONG IT TAKES TO GET MAIL FROM ONE PLACE TO ANOTHER. I POSTED A JOURNAL TO SYDNEY (160 KM) IN DECEMBER 1991 AND IT ARRIVED IN JANUARY 1993. CAN ANYONE BEAT THAT?

CONDOLENCES: Our condolences go to the family and friends of Neville Hughes who passed away recently. Neville, although in his seventies was a very active VZ user and will be sadly missed.

FAREWELL HARRY & VEE ZEDD DOWN UNDER

After 4.5 years as Editor of Vee Zedd Down Under, Harry Huggins has decided to retire from editing with next issue being the last. On behalf of all VZ users I wish Harry a full recovery from his recent illness and all the best for the future. Also a **BIG THANK YOU** for a job well done.

IS THE END OF THE VZ NEAR?

As there were no takers for Harry's job it was decided to merge Vee Zed Down Under with the Hunter Valley VZ Journal which will make it the last VZ publication left in Australia. It is unfortunate that both magazines survival has depended solely on their Editors.

For a couple years now I have been producing the Journal from start to finish on my own and the last 18 months in my bedroom under difficult circumstances as my brother is a shiftworker. My intention is to keep producing the Journal for as long as I can and while the demand is there.

Picking up extra members from Vee Zed Down Under will make the Journal a more viable proposition. The support of remaining members via their subscriptions and articles is vital to the Journal's continued existence.

Our local club meetings have become IBM orientated with not a VZ in sight unless the meeting is at my place. It has been suggested that the Journal cater for other computer systems like the IBM. Personally I cannot see the sense in it as computers like the IBM have massive hardware, software, and technical support, etc, etc.

Let us not forget that originally VZ User Groups and publications like the VZ 200-300 Hunter Valley Journal, VZ Down Under were formed to give additional support to that available from Dick Smith. Now more than ever the VZ needs all the support it can get.

NOTE: Work has started on my house and should be to lock up stage by the time you get this issue and is one reason why this issue is late, Ed.

WANTED TO BUY 64K RAM PACK — CONTACT

Stephen Gayst 2/118-122 Pacific Hwy Roseville 2069 (02) 416 6714

VZ SYSTEM FOR SALE — CONTACT

Jack Shearsmith 95 Modillion Ave Riverton 6148

1 off VZ, power supply and 16K Mem Expansion
1 off Disk Drive, Disk Controller and power supply
1 off Word Pro Cartridge

All necessary manuals, several Dick Smith books on programs, technical manual, and all the VZ User magazines and several disks.

Cost: about $150.00 plus postage

FUNCTION 25 - GET ALTERNATE SUBROUTINE ADDRESS.

```
    PARAMETERS: M1% = 25
                M3% = CALL MASK.
RETURN VALUES: M1% = ERROR FLAG (-1 IF ERROR)
               M3% = CALL MASK
               M4% = SUBROUTINE ADDRESS.
```

DESCRIPTION: THIS FUNCTION RETURNS THE SUBROUTINE ADDRESS WHICH IS IDENTIFIED BY THE SPECIFIED CALL MASK. THE MASK DEFINITION WAS DESCRIBED IN FUNCTON 24.

FUNCTION 28 - SET MOUSE INTERRUPT RATE.

```
    PARAMETERS: M1% = 28
                M2% = POLLING RATE
RETURN VALUES: NONE.
```

DESCRIPTION: THIS FUNCTION SETS THE NUMBER OF TIMES THE MOUSE IS POLLED EACH TIME AN INTERRUPT OCCURS. NOTE THAT THIS IS NOT DIRECTLY EQUIVALENT TO THE MICROSOFT FUNCTION 28.

FUNCTION 30 - GET CRT PAGE NUMBER.

```
    PARAMETERS: M1% = 30
RETURN VALUES: M2% = CRT PAGE OF CURRENT CURSOR DISPLAY.
```

DESCRIPTION: THIS FUNCTION CURRENTLY RETURNS PAGE 0 AS ONLY MODE 0 IS SUPPORTED.

FUNCTION 31 - DISABLE MOUSE DRIVER.

```
    PARAMETERS: M1% = 31
RETURN VALUES: M1% = ERROR STATUS (-1 IF ERROR)
```

DESCRIPTION: THIS FUNCTION DISABLES THE MOUSE INTERRUPT ROUTINE. NOTE THAT THE DRIVER IS NOT UNLOADED FROM MEMORY.

FUNCTION 32 - ENABLE MOUSE DRIVER.

```
    PARAMETERS: M1% = 32
RETURN VALUES: AS PER FUNCTION 0.
```

DESCRIPTION: THIS FUNCTION ENABLES THE MOUSE DRIVER INTERRUPT ROUTINE.

FUNCTION 35 - GET LANGUAGE NUMBER.

```
    PARAMETERS: M1% = 35
RETURN VALUES: M2% = CURRENT LANGUAGE NUMBER.
```

DESCRIPTION: THIS FUNCTION RETURNS THE NUMBER OF THE LANGUAGE CURRENTLY SET IN THE DRIVER. (0 = ENGLISH).

FUNCTION 36 - GET DRIVER VERSION AND MOUSE TYPE.

```
    PARAMETERS: M1% = 36
RETURN VALUES: M2% = DRIVER VERSION.
               M3% = MOUSE TYPE IN UPPER BYTE.
```

DESCRIPTION: THIS FUNCTION RETURNS THE VERSION OF THE DRIVER INSTALLED AND THE TYPE OF MOUSE. NOTE THAT THIS FUNCTION IS NOT DIRECTLY EQUIVALENT TO THE MICROSOFT FUNCTION 36.

FUNCTION 38 - GET MAXIMUM VIRTUAL COORDS.

```
    PARAMETERS: M1% = 38
RETURN VALUES: M2% = MOUSE ENABLED FLAG.
               M3% = MAXIMUM HORIZONTAL COORD.
               M4% = MAXIMUM VERTICAL COORD.
```

DESCRIPTION: THIS FUNCTION RETURNS THE MAX HORIZONTAL AND VERTICAL COORDS FOR THE CURRENT SCREEN MODE.

FUNCTION 39 - GET SCREEN/CURSOR MASKS AND MOTION COUNTS.

```
    PARAMETERS: M1% = 39
RETURN VALUES: M1% = SCREEN MASK
               M2% = CURSOR MASK
               M3% = HORIZONTAL MOTION COUNT.
               M4% = VERTICAL MOTION COUNT.
```

DESCRIPTION: THIS FUNCTION RETURNS THE SCREEN AND CURSOR MASKS (AS DESCRIBED IN FUNCTION 10) AND THE CURRENT MOTION COUNTS (NOTE THAT THESE ARE NOT RESET AFTER THIS FUNCTION IS CALLED).

FUNCTION 40 - SET VIDEO MODE.

```
    PARAMETERS: M1% = 40
               M3% = VIDEO MODE
RETURN VALUES: M3% = ERROR FLAG. (0 = SUCCESS)
```

DESCRIPTION: THIS FUNCTION SETS THE DRIVERS VIDEO MODE TO THE SPECIFIED MODE, IF IT IS VALID.

FUNCTION 41 - LIST VIDEO MODES.

```
    PARAMETERS: M1% = 41
               M3% = FIND MODE (0 = FIND FIRST)
RETURN VALUES: M3% = FOUND VIDEO MODE NUMBER
               M4% = POINTER TO STRING DESCRIPTION.
```

DESCRIPTION: THIS FUNCTION LISTS ALL VIDEO MODES SUPPORTED BY THE DRIVER. ONCE ALL MODES HAVE BEEN LISTED M3% = -1. M4% POINTS TO A 0 BYTE TERMINATING STRING DESCRIPTION OF THE FOUND VIDEO MODE.

FUNCTION 49 - GET MIN/MAX COORDS.

```
    PARAMETERS: M1% = 49
RETURN VALUES: M1% = MIN HORIZONTAL POS.
               M2% = MIN VERTICAL POS.
               M3% = MAX HORIZONTAL POS.
               M4% = MAX VERTICAL POS.
```

DESCRIPTION: THIS FUNCTION RETURNS THE MIN AND MAX COORDS FOR THE CURRENT VIDEO MODE, AS SET BY FUNCTIONS 7 & 8. IF THESE FUNCTIONS HAVE NOT BE CALLED THEN THE ABSOLUTE MIN AND MAX COORDS ARE RETURNED.

FUNCTION 50 - GET ACTIVE ADVANCED FUNCTIONS.

```
    PARAMETERS: M1% = 50
RETURN VALUES: M1% = ACTIVE FUNCTIONS FLAG.
```

DESCRIPTION: THIS FUNCTION RETURNS 16 FLAGS (BITS) THAT INDICATE ACTIVE ADVANCED FUNCTIONS. THE MOST SIGNIFICANT BIT OF M1% IS SET IF FUNCTION 37 IS AVAILABLE OTHERWISE IT IS RESET. THE LEAST SIGNIFICANT BIT OF M1% CORRESPONDS TO FUNCTION 52 AND IS SET IF FUNCTION 52 IS AVAILABLE.

MICROSOFT MOUSE FUNCTIONS NOT SUPPORTED BY THE VZ BUS MOUSE DRIVER (V1.04)

FUNCTION	DESCRIPTION
9	SET GRAPHICS CURSOR BLOCK.
13	LIGHT PEN EMULATION ON.
14	LIGHT PEN EMULATION OFF.
15	SET MICKEY/PIXEL RATIO.
16	CONDITIONAL OFF.
19	SET DOUBLE SPEED THRESHOLD.
26	SET MOUSE SENSITIVITY.
27	GET MOUSE SENSITIVITY.
29	SET CRT PAGE NO.
33	SOFTWARE RESET.
34	SET LANGUAGE FOR MESSAGES
37	GET GENERAL DRIVER INFORMATION.
42	GET CURSOR HOT SPOT.
43	LOAD ACCELERATION CURVES.
44	READ ACCELERATION CURVES.
45	SET/GET ACTIVE ACCELERATION CURVE.
47	MOUSE HARDWARE RESET.
48	SET/GET BALLPOINT.
51	GET SWITCH SETTINGS.
52	GET MOUSE.INI

IMPORTANT NOTES:

(1) FUNCTIONS 17, 18, 46 ARE INTERNAL FUNCTIONS AND SHOULD NOT BE CALLED DIRECTLY BY ANY APPLICATION.

(2) ALTHOUGH FUNCTIONS EQUIVALENT TO THE MICROSOFT MOUSE DRIVER HAVE BEEN PROVIDED, NOT ALL PASSED PARAMETERS AND RETURN VALUES ARE THE SAME.

(3) APPLICATION PROVIDED SUBROUTINES CALLED BY THE MOUSE DRIVER MUST PRESERVE ALL REGISTERS USED. UPON ENTRY TO A SUBROUTINE THE REGISTER VALUES ARE:-

    E - BUTTON REGISTER.
    A - CONDITION WHICH CAUSED SUBROUTINE CALL.
    B - INTERRUPT LOOP COUNT.
    C - MOUSE PORT NUMBER.
    H - CURRENT HORIZONTAL POS.
    L - CURRENT VERTICAL POS.
   IX - POINTER TO MOUSE VARIABLES.

USING THE MOUSE DRIVER FUNCTIONS:-

(A) WITH BASIC PROGRAMS

THIS IS EASY TO DO AS THE NEW COMMAND MOUSE() IS PROVIDED WHEN THE MOUSE DRIVER IS LOADED. REFER TO PART 1 FOR DETAILS OF THIS FUNCTION.

(B) WITH MACHINE CODE PROGRAMS

TO USE THE MOUSE DRIVER WITH M/C PROGRAMS AN ALTERNATE CALLING METHOD HAS BEEN PROVIDED. AS THE DRIVER IS RELOCATABLE, DIRECT ADDRESSING CANNOT BE USED TO CALL THE FUNCTIONS. AS WITH MY LAST PROJECT, KSCAN, IMPORTANT POINTERS ARE STORED IN THE COMMUNICATIONS REGION. THESE ARE:-

```
MPTR 31278-79: THIS POINTS TO THE DRIVER.
MFUN 31280-82: THIS AIDS INTERNAL INDIRECT FUNCTION CALLS.
MCAL 31283-85: THIS POINTS TO THE FUNCTION CALLING ROUTINE.
PREG 31286-87: THIS POINTS TO THE MOUSE VARIABLES.
```

NOTE: THESE ADDRESSES CLASH WITH THE FIND ROUTINE BUT NOT KSCAN.

THE POINTER OF MAIN INTEREST IS MCAL. THIS ALLOWS A M/C ROUTINE TO CALL ANY OF THE MOUSE FUNCTIONS. THIS IS DONE AS FOLLOWS:-

```
LD IX, M1% (THE MOUSE FUNCTION NUMBER)
LD BC, M2% (THE FIRST PARAMETER)
LD DE, M3% (THE SECOND PARAMETER)
LD HL, M4% (THE THIRD PARAMETER)
CALL MCAL
```

GETTING AND RUNNING THE DRIVER:-

THE MOUSE DRIVER ACCEPTS COMMAND LINE OPTIONS. CURRENTLY ONLY TWO ARE SUPPORTED, THESE ARE -P AND -A. THE -A OPTION ALLOWS THE USER TO SPECIFY THE DESTINATION ADDRESS OF THE DRIVER. THE -P OPTION ALLOWS THE USER TO SPECIFY THE MOUSE PORT. NOTE THAT IF MORE THAN ONE OPTION IS SPECIFIED THEY MUST BE SEPERATED BY A SPACE.

EXAMPLES:

(I) BRUN"MD1.04" OR BRUN"MD1.04":

THIS INSTALLS THE DRIVER AT TOP OF MEMORY AND SELECTS THE DEFAULT MOUSE PORT (PORT 2).

(II) BRUN"MD1.04":-A49152 OR BRUN"MD1.04":-A49152:

THIS INSTALLS THE DRIVER AT ADDRESS 49152 AND SELECTS THE DEFAULT MOUSE PORT.

(III) BRUN"MD1.04":-P20 OR BRUN"MD1.04":-P20:

THIS INSTALLS THE DRIVER AT THE TOP OF MEMORY AND SELECTS PORT 20 AS THE MOUSE PORT.

(IV) BRUN"MD1.04":-P100 -A50000 OR BRUN"MD1.04":-A50000 -P100

THIS INSTALLS THE DRIVER AT ADDRESS 50000 AND SELECTS PORT 100 AS THE MOUSE PORT.

MISCELLANEOUS NOTES:-

(1) IN PART 1, I MENTIONED THAT THE MOUSE DRIVER WAS VERSION 1.03. IT HAS SINCE BEEN UPGRADED TO 1.04

(2) A NEW BOOK IS AVAILABLE AT ANGUS AND ROBERTSON BOOKSHOPS WHICH DESCRIBES THE FIRST 36 MOUSE FUNCTIONS. IT IS THE PROGRAMMERS QUICK REFERENCE MS-DOS EXTENSIONS BY RAY DUNCAN FROM MICROSOFT PRESS AND COSTS $10-95.

PROBLEMS WITH THE MOUSE DRIVER (V1.04):-

THESE ARE NOT SERIOUS JUST ANNOYING:-

(1) THE DRIVER OVERRIDES ANY OTHER INSTALLED INTERRUPT ROUTINE (INCLUDING KSCAN).

(2) THERE IS CURRENTLY NO WAY TO TELL IF THE DRIVER IS LOADED.

(3) DEPENDING ON THE NUMBER OF TIMES THE MOUSE IS POLLED PER INTERRUPT, THE INBUILT BASIC EDITOR CAN SLOW DOWN DRAMATICALLY.

SOLUTIONS:

THE FIRST TWO PROBLEMS WILL BE RESOLVED IN A FUTURE UPGRADE. ONCE THEY ARE RESOLVED, THE THIRD PROBLEM MAY BE SOLVED BY USING KSCAN RATHER THAN THE DEFAULT KEY SCANNING ROUTINE.

FUTURE SUGGESTIONS:

MY ARTICLE AND GARY BULLEY'S ARTICLE ARE ONLY TWO SOLUTIONS TO THE PROBLEM OF CONNECTING A MOUSE TO THE VZ. BOTH WASTE CPU TIME POLLING THE MOUSE FOR ACTIVITY, MAKING A DECISION AND THEN ACTING UPON THAT DECISION. WHAT IS REALLY REQUIRED IS A HARDWARE "ADDON" - A MOUSE CONTROLLER. THIS SHOULD MONITOR THE MOUSE, MAINTAIN ALL MOUSE VARIABLES, AND MAKE A DECISION ON AN ACTION TO BE TAKEN. IT SHOULD THEN INFORM THE CPU WHICH WILL PERFORM THE REQUIRED ACTION, SUCH AS MOVING THE MOUSE CURSOR

MOUSE DRIVER SOURCE CODE:

UNFORTUNATELY THE SOURCE CODE LISTING IS QUITE LENGTHY AND IS SIMPLY TOO LONG TO PUBLISH IN THE JOURNAL - SORRY ABOUT THAT. HOWEVER, BOTH THE SOURCE CODE AND EXECUTABLE ARE AVAILABLE FROM THE EDITOR, JOE LEON. PLEASE NOTE THAT ONLY AN EDITOR ASSEMBLER WITH THE DISKOPS6 PATCH CAN REGENERATE THE DRIVER.

DISKOPS6 PROVIDES AN OPTION TO LINK OBJECT CODE WHICH IS STORED IN DIFFERENT FILES. IT REQUIRES THE 64K EXPANSION MODULE. IF YOU WOULD LIKE A COPY OF THIS (YET ANOTHER!) VERSION OF EDAS IT CAN ALSO BE OBTAINED VIA THE EDITOR, ADDRESS BELOW.

JOSEPH P LEON 33 TIGHES TERRACE TIGHES HILL 2297 (049) 69 2399

EDITOR'S NOTE: THE OBJECT AND SOURCE CODES FILES FOR THE BUS MOUSE DRIVER ARE SHOWN BELOW.

```
    B:MD1.04    01 00 7AE9 840D 0924 - MOUSE DRIVER

    W:DRIVER28 02 0D A813 E11F 390C -  713 LINES
    W:MBAS1.0  0A 01 A813 B8AB 1098 -  228    "
    W:MASS1.0  0C 03 A813 F966 5153 -  974    "
                                            ----
                                    1915 LINES SOURCE CODE
```

AS YOU CAN SEE THAT IS A LOT OF SOURCE CODE TO ENTER. IT WOULD TAKE ABOUT 15.5 PAGES AT 124 LINES PER PAGE AND WOULD TAKE ANOTHER 3 ISSUES TO PUBLISH. ALSO AS LESLIE ALREADY MENTIONED DISKOPS6, A 64K VERSION OF EDITOR ASSEMBLER AND A 64K MEMORY EXPANSION IS REQUIRED.

MOUSE DRIVER TEST PROGRAM

```
10 CLS:PRINT"TOM  =";:PRINT PEEK(30897)+256*PEEK(30898)
20 PRINT"MPTR =";:PRINT PEEK(31278)+256*PEEK(31279)
30 PRINT"PREG =";:PRINT PEEK(31286)+256*PEEK(31287)
40 PRINT"MFUN =";
45 PRINT PEEK(31280);:PRINT PEEK(31281)+256*PEEK(31282)
50 PRINT"MCAL =";
55 PRINT PEEK(31283);:PRINT PEEK(31284)+256*PEEK(31285)
60 PRINT"INTR =";
65 PRINT PEEK(30845);:PRINT PEEK(30846)+256*PEEK(30847)
```

MOUSE DRIVER EXAMPLE PROGRAM

```
010 REM MOUSE DRIVER EXAMPLE
020 CLS
030 REM DISPLAY SELECTION MENU
040 PRINT"PLEASE SELECT:-"
050 PRINT
060 PRINTTAB(10);"1. DIR"
070 PRINTTAB(10);"2. STATUS"
080 PRINTTAB(10);"3. RENAME A FILE"
090 PRINTTAB(10);"4. FORMAT A DISK"
095 PRINT
100 MOUSE(0,0,0,0) 'RESET MOUSE
110 MOUSE(4,0,28,2) 'SET MOUSE POSITION
120 MOUSE(7,0,28,28) 'LIMIT CURSOR TO CURRENT COLUMN
130 MOUSE(8,0,2,5) 'LIMIT CURSOR TO ROWS 2 - 5
140 MOUSE(28,50,0,0) 'REDUCE MOUSE SENSITIVITY
150 MOUSE(1,0,0,0) 'DISPLAY MOUSE CURSOR
200 REM WAIT FOR MOUSE BUTTON PRESS
210 MOUSE(3,M2%,M3%,M4%) 'GET BUTTON PRESS INFO
220 REM CHECK FOR LEFT BUTTON PRESS
230 IF (M2%AND1)<>1,210
235 RS%=M4%:MOUSE(2,0,0,0) 'HIDE MOUSE CURSOR
240 REM PRINT THE STRING OF THE ROW SELECTED.
250 IF RS%=2,PRINT"DIR SELECTED"
260 IF RS%=3,PRINT"STATUS SELECTED"
270 IF RS%=4,PRINT"RENAME SELECTED"
280 IF RS%=5,PRINT"FORMAT SELECTED"
```

VZ BUS MOUSE INPUT PORT

AS MENTIONED PREVIOUSLY, YOU'LL HAVE TO MODIFY GARY BULLEY'S 8 BIT INPUT PORT OR BUILD THE ONE FEATURED WITH THIS ARTICLE. THE BUS MOUSE REQUIRES NO MODIFICATION AT ALL. IF YOU ELECT TO MODIFY GARY'S 8 BIT INPUT PORT FROM ISSUE # 30, PAGE 11 THEN TWO CHANGES ARE REQUIRED.

1) REMOVE DB15 PLUG AND WIRE UP DB9 MALE PLUG AS SHOWN ON NEXT PAGE. BE CAREFUL TO GET THE RIGHT SIGNAL TO THE RIGHT PIN ACCORDING TO DB9 DIAGRAM.

2) D7 ON INPUT SIDE OF 74LS245 (PIN 2) HAS TO BE GROUNDED. YOU CAN LEAVE CAPACITOR IN PLACE AS IT WILL BE EASIER THAN REMOVING IT.

NOTE: LEAVE PIN 19 OF 74LS245 CONNECTED TO PIN 10 OF 74LS138.

THE BUS MOUSE INPUT PORT BELOW WAS DESIGNED TO GO INSIDE A VZ200 OR VZ300 WITH BUS CONNECTIONS COMING FROM THE 30 WAY I/O EXPANSION CONNECTOR. I HAVE BUILT IT, BUT HAVEN'T GOT AROUND TO INSTALLING IT IN MY VZ AS YET. PINOUTS OF BOTH IC'S, DB9 AND 30 WAY EDGE CONNECTOR ARE GIVEN TO HELP INTENDING CONSTRUCTORS.

INPUT PORT SELECTION:

THIS HAS BEEN LEFT TO THE USER TO SELECT. YOU CAN USE I/O PORT ADDRESS SHOWN BY DOTTED LINE BETWEEN 74LS138, PIN 15 AND 74LS245, PIN 19. THE SQUARE MARKERS ON 74LS138 SHOW THE I/O ADDRESSES AVAILABLE. OR IF YOU WISH YOU CAN SELECT AN I/O DECODER FROM TECHNICAL DATA SHEET # 4, ISSUE 41, PAGE 18.

CONSTRUCTION SHOULD CAUSE NO PROBLEMS AS IT IS A FAIRLY SIMPLE PROJECT. LESLIE HAS PROVIDED A MOUSE DRIVER AND IT IS UP TO PROGRAMMERS TO ACCESS ITS FUNCTIONS FROM THEIR OWN PROGRAMS, ED.

BUSMOUSE   VZ 200/300 BUS MOUSE INPUT PORT BY JOE LEON   19-12-1992
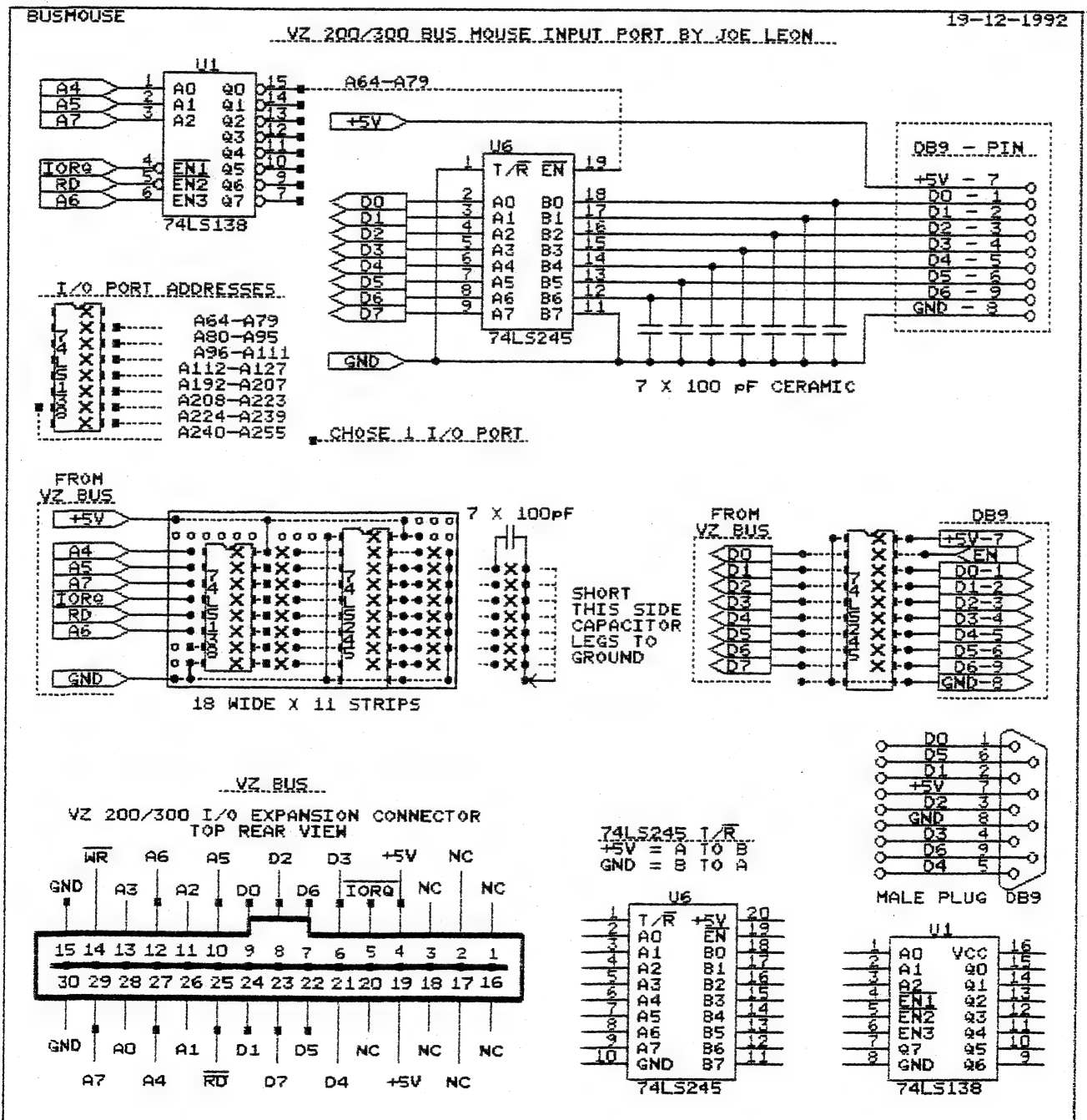
```
00001 ;CHECK DISK ROUTINE
00002 ;SELECT DRIVE
00003 ;18:02:92
00004 ;PRINT USR(0) TO RERUN
00005 ;ORG.: 9000H
00006          LD    HL,ST
00007          LD    (788EH),HL
00008 ST       CALL  01C9H
00009          LD    HL,MES1
00010          CALL  2B75H
00011 A1       CALL  0049H
00012          CP    0DH
00013          JR    Z,A1A
00014          CP    32H
00015          JR    Z,TWO
00016          CP    31H
00017          JR    NZ,A1
00018          LD    A,10H
00019          JR    T1
00020 TWO      LD    A,80H
00021 T1       PUSH  AF
00022          DI
00023          CALL  4008H
00024          LD    A,(IY+20)
00025          OR    A
00026          JR    Z,T2
00027          LD    B,A
00028          CALL  403EH
00029 T2       CALL  400BH
00030          POP   AF
00031          LD    (IY+0BH),A
00032 A1A      LD    (IY+11H),0
00033          LD    (IY+12H),0
00034          DI
00035          CALL  3450H
00036          CALL  4008H
00037          LD    BC,0032H
00038          CALL  4038H
00039 A2       CALL  4035H
00040          OR    A
00041          JR    Z,A3
00042          LD    HL,MES2
00043          JP    END
00044 A3       INC   (IY+11H)
00045          LD    A,0FH
00046          CP    (IY+11H)
00047          JR    NZ,A2
00048          CALL  4035H
00049          OR    A
00050          JP    NZ,S1
00051          LD    (IY+11H),0
00052          LD    (IY+12H),1
00053          LD    L,(IY+31H)
00054          LD    H,(IY+32H)
00055          LD    E,(IY+34H)
00056          LD    D,(IY+35H)
00057          LD    BC,0050H
00058          LDIR
00059 A4       DI
00060          CALL  4035H
00061          OR    A
00062          JP    NZ,ERR
00063 A4A      INC   (IY+11H)
00064          LD    A,10H
00065          CP    (IY+11H)
00066          JR    NZ,A4
00067          LD    (IY+11H),0
00068          INC   (IY+12H)
00069          LD    A,28H
00070          CP    (IY+12H)
00071          JR    NZ,A4
00072          LD    L,(IY+31H)
00073          LD    H,(IY+32H)
00074          PUSH  HL
00075          LD    (HL),0
00076          POP   DE
00077          PUSH  DE
00078          INC   DE
00079          LD    BC,007FH
00080          LDIR
00081          POP   DE
00082          LD    L,(IY+34H)
00083          LD    H,(IY+35H)
00084          LD    BC,0050H
00085          LDIR
00086          CALL  WP
00087          LD    (IY+11H),0FH
00088          LD    (IY+12H),0
00089          CALL  4032H
00090          OR    A
00091          JR    Z,A5
00092 S1       LD    HL,MES3
00093          JP    END
00094 A5       LD    L,(IY+31H)
00095          LD    H,(IY+32H)
00096          LD    DE,0
00097          LD    C,4EH
00098 A6       LD    B,8
00099          LD    A,(HL)
00100 A7       RRC   A
00101          JR    C,A8
00102          INC   DE
00103 A8       DJNZ  A7
00104          INC   HL
00105          DEC   C
00106          JR    NZ,A6
00107          PUSH  DE
00108          PUSH  DE
00109          POP   HL
00110          CALL  0FAFH
00111          LD    HL,MES4
00112          CALL  2B75H
00113          POP   HL
00114          PUSH  HL
00115          SRL   H
00116          RR    L
00117          SRL   H
00118          RR    L
00119          SRL   H
00120          RR    L
```

```
00121          CALL  0FAFH              00167       EI
00122          LD    A,2EH              00168       LD    HL,WP1
00123          CALL  033AH              00169       CALL  2B75H
00124          POP   HL                 00170       LD    HL,WP2
00125          LD    A,7                00171       CALL  2B75H
00126          AND   L                  00172 FA    LD    A,(7AAFH)
00127          INC   A                  00173       OR    A
00128          LD    B,A                00174       JR    NZ,FA
00129          LD    HL,0FF83H          00175 K2    CALL  0049H
00130          LD    DE,007DH           00176       CP    0DH
00131 A9       ADD   HL,DE              00177       JR    NZ,K2
00132          DJNZ  A9                 00178       DI
00133          CALL  0FAFH              00179       CALL  3450H
00134          LD    HL,MES5            00180       CALL  4008H
00135 END      CALL  2B75H             00181       LD    BC,0032H
00136          CALL  400BH             00182       CALL  4038H
00137          JP    1A19H              00183       RET
00138 ERR      LD    L,(IY+34H)         00184 MES1  DEFB  1FH
00139          LD    H,(IY+35H)         00185 *           CHECK DISK*
00140          LD    A,(IY+12H)         00186       DEFB  0DH
00141          DEC   A                  00187 *     WRITTEN BY D.MITCHELL*
00142          SLA   A                  00188       DEFB  0DH
00143          LD    E,A                00189 *     LAST UPDATE : 18:02:92*
00144          LD    D,0                00190       DEFB  0DH
00145          LD    A,(IY+11H)         00191 WP2   DEFB  0DH
00146          CP    8                  00192 *     PRESS RETURN OR 1 OR 2*
00147          CCF                      00193       DEFB  0DH
00148          ADC   HL,DE              00194       NOP
00149          AND   7                  00195 MES2  EQU   $
00150          INC   A                  00196 *ERROR IN DIRECTORY SECTORS
00151          LD    B,A                00197 * TRY  REFORMATTING*
00152          LD    C,(HL)             00198       DEFW  000DH
00153          RLC   C                  00199 MES3  EQU   $
00154 A13      RRC   C                  00200 *ERROR IN STATUS SECTOR TRY
00155          DJNZ  A13                00201 *        REFORMATTING*
00156          SET   0,C                00202       DEFW  000DH
00157          LD    B,A                00203 MES4  EQU   $
00158          RRC   C                  00204 * SECTORS FREE      *
00159 A14      RLC   C                  00205       NOP
00160          DJNZ  A14                00206 MES5  EQU   $
00161          LD    (HL),C             00207 *K FREE*
00162          JP    A4A                00208       DEFW  0D0DH
00163 WP       IN    A,(13H)            00209       NOP
00164          BIT   7,A                00210 WP1   EQU   $
00165          RET   Z                  00211       DEFB  0DH
00166          CALL  400BH             00212 *REMOVE WRITE PROTECT LABEL*
                                        00213       NOP
```

THIS IS AN UPDATE OF ONE MY SMALL ROUTINES WHICH WAS PREVIOUSLY
PUBLISHED IN ISSUE # 24, MAY/JUNE 1989. I USE CHECK DISK WHEN I GET
AN INPUT/OUTPUT ERROR AFTER I HAVE INITIALIZED A DISK. THE PROGRAM
READS EACH SECTOR OF THE DISK AND CHECKS FOR ERRORS. THIS PROGRAM
DOES NOT STOP THE ERROR BUT IT WILL RETRIEVE SOME DISKS FROM THE SCRAP
HEAP. THE PROGRAM WILL REDUCE THE STORAGE AREA ON THE DISK IF IT
CANNOT FIND TRACKS OR SECTORS, THE AMOUNT OF FREE SPACE LEFT ON THE
DISK (STATUS) WILL BE PRINTED TO THE SCREEN AT THE END OF THE PROGRAM.

    WITH IBM & COMPATABLE COMPUTERS A CHECK IS MADE OF DISKS AFTER
THEY ARE FORMATTED. IF SECTORS CANNOT BE FOUND, WHOLE TRACKS ARE
LOCKED OFF FROM THE USER.

WITH THE VZ A CHECK IS ALSO MADE BUT WHEN ERRORS ARE FOUND IT PRINTS THE DREADED INPUT/OUTPUT ERROR.

CHECK DISK DOES NOT LOCK OFF WHOLE TRACKS JUST THE SECTORS THAT THE ERROR OCCURRED IN.

CHECK DISK CAN ALSO BE USED ON A DISK THAT HAS PROGRAMS STORED ON IT AS CHKDSK PLACES THE TRACK MAP INTO MEMORY AND READS THE DISK THEN SAVES THE MAP TO DISK.

IF YOU SAVE A PROGRAM AND GET AN I/O ERROR THEN ERASE THAT PROGRAM AND BRUN CHKDSK THIS SHOULD REPAIR THE ERROR.

AFTER THE CHKDSK PROGRAM HAS RUN A MESSAGE WILL APPEAR ON THE SCREEN PRESSING RETURN THE PROGRAM WILL CHECK THE DISK IN THE DRIVE THAT CHKDSK WAS LOADED FROM.

PRESSING THE 1 KEY WILL CHECK THE DISK IN DRIVE 1 AND PRESSING 2 KEY WILL CHECK THE DISK IN DRIVE 2.

WHEN THE 1 OR 2 KEYS ARE PRESSED THE DRIVE THAT CHKDSK WAS LOADED FROM WILL RESET TO TRACK ZERO BEFORE SELECTING THE RELEVANT DRIVE. AGAIN THIS IS TO TRY AND STOP THE BASHING OF THE DRIVE HEADS.

CHECKDISK 2 COMMANDS:

```
RETURN - WILL CHECK DISK FROM DRIVE CHKDSK2 WAS BRUN FROM.
     1 - WILL CHECK DISK IN DRIVE 1.
     2 - WILL CHECK DISK IN DRIVE 2.
         MAKE SURE THAT DRIVE 2 IS ACTIVATED OR VZ COULD HANG UP.
  QUIT - OPEN DRIVER DOOR, PRESS RETURN THEN PRESS (-) MINUS KEY
         AFTER WHICH AN ERROR MESSAGE WILL BE PRINTED OUT AND CAN
         BE IGNORED.
RE-ENTER - PRINT USR(0)
```

PARK UPDATE BY DAVE MITCHELL

```
00001 ;PARK ROUTINE FOR TWO DRIVES    00023      PUSH AF
00002 ;WRITTEN BY D.MITCHELL          00024      CALL 4008H
00003 ;LAST UPDATE : 20:06:92         00025      POP  AF
00004      DI                         00026      LD   B,A
00005      LD   (IY+0),0              00027      CALL 403BH
00006      LD   A,(IY+11)            00028      CALL 400BH
00007      CP   80H                   00029 END  EI
00008      JR   NZ,D1                 00030      LD   HL,MES
00009      LD   (IY+0),2              00031      CALL 2B75H
00010      LD   A,(IY+20)            00032      LD   A,(IY+0)
00011      OR   A                     00033      OR   A
00012      JR   Z,D1                  00034      JR   Z,A1
00013      PUSH AF                    00035      LD   (IY+0),0
00014      CALL 4008H                 00036      LD   HL,TWO
00015      POP  AF                    00037      JR   A2
00016      LD   B,A                   00038 A1   LD   HL,ONE
00017      CALL 403EH                 00039 A2   CALL 2B75H
00018      CALL 400BH                 00040      JP   1A19H
00019 D1   LD   (IY+11),10H          00041 MES  DEFB 1FH
00020      LD   A,27H                 00042 *       DRIVE PARK ROUTINE*
00021      SUB  (IY+20)              00043      DEFB 0DH
00022      JR   Z,END                 00044 *       WRITTEN BY D.MITCHELL*
                                      00045      DEFB 0DH
```

```
00045       DEFB 0DH              00052       DEFB 0DH
00046 *     LAST UPDATE: 20.06.92*  00053 ONE  EQU  $
00047       DEFB 0DH              00054 * DRIVE 1 PARKED AT TRACK*
00048       NOP                   00055 * 39*
00049 TWO   EQU  $                00056       NOP
00050 * DRIVE 2 PARKED AT TRACK*  00057       NOP
00051 * 00*
```

SAVE YOUR OBJECT CODE AS PARK2 SO IT WONT GET MIXED UP WITH OTHER VERSIONS AND WHEN YOU RUN PARK2 YOU'LL SEE ONE OF TWO DISPLAYS SHOWN BELOW DEPENDING FROM WHICH DRIVE YOU LOADED PARK2 FROM.

```
        DRIVE PARK ROUTINE              DRIVE PARK ROUTINE
       WRITTEN BY D.MITCHELL           WRITTEN BY D.MITCHELL
        LAST UPDATE:  20.6.92           LAST UPDATE:  20.6.92
      DRIVE 2 PARKED AT TRACK 00      DRIVE 1 PARKED AT TRACK 39
      DRIVE 1 PARKED AT TRACK 39      READY
READY
```

EDITORS NOTE: AS A REGULAR VZ USER I USE PARK VERY FREQUENTLY TO RESET MY VZ WITHOUT BANGING THE DRIVE HEADS AND DIMINISHING IT'S LIFE EXPECTANCY. MOST OF MY BASIC PROGRAMS HAVE A "QUIT & PARK" OPTION WHICH I USE TO EXIT THE PROGRAM. THEN THE VZ GETS TURNED OFF OR THE RESET BUTTON IS PRESSED TO CLEAR MEMORY AND I LOAD ANOTHER PROGRAM. EITHER WAY THERE'S NO MORE HEAD BANGING. HAPPY PARKING!

CONVERTING SOURCE CODE UPDATE

DAVE MITCHELL HAS MODIFIED HIS EDITOR ASSEMBLER SO IT WILL ALSO LOAD W:SOURCE & A:SOURCE CODE FILES WITHOUT MODIFICATION.

AT LONG LAST IN-COMPATIBILITY IS NO LONGER A PROBLEM FOR ONE EDITOR ASSEMBLER AND I HOPE OTHER AUTHORS WILL FOLLOW SUIT. BELOW ARE THE COMMANDS USED TO LOAD THE VARIOUS SOURCE CODE FILES.

TL:FILENAME WILL LOAD DM(S) & MH(S) SOURCE CODE FILES.
TM:FILENAME WILL LOAD RH(A) & LM(W) SOURCE CODE FILES.

NOTE: THE COMMAND TM: IS NORMALLY USED TO MERGE SOURCE CODE FILES, ED.

EXT DOS V12.1 UPDATE

THE FOLLOWING COMMANDS AND ENHACEMENTS HAVE BEEN MADE TO EXT12.1 WHICH IS NOW KNOWN AS EXT12.2.

DIS? - NUMBER OF BYTES AND FILES HAS BEEN ADDED.
LDIS? - AS DIS? EXCEPT ALL OUTPUT GOES TO THE PRINTER.
EREL"FILENAME",XXXX - RELOCATES END OF FILE

THIS COMMAND IS SIMILIAR TO REL"FILENAME", EXCEPT IT END RELOCATES A FILE. THE FILENAME AND NEW END ADDRESS IS REQUIRED. A NEW START ADDRESS IS AUTOMATICALLY CALCULATED AND BOTH THE NEW START AND END ADDRESSES ARE PLACED INTO THE DISK DIRECTORY. SEE EXAMPLE BELOW.

SAMPLE FILE - W:MOUSE3 18 02 D861 FE00 259F - R. HARRISON WP FILE

COMMAND: - EREL"MOUSE3",D000 - USE HEX NUMBERS ONLY.

RESULT: - W:MOUSE3   18 02 AA61 D000 259F - D. MITCHELL WP FILE

MOST OF YOU WILL NEVER USE THE EREL COMMAND UNLESS YOU GET RUSSELL HARRISON WP FILE/S AND YOU ONLY HAVE DAVE MITCHELL'S WP. A RH.WP FILE WOULD OVER-WRITE DAVE MITCHELL'S WP RESERVED MEMORY AREA ABOVE D000 AND HANG UP PROGRAM. EREL WILL SOLVE THAT PROBLEM NEATLY.

I HAVE BEEN ASKED TO CONTRIBUTE A SERIES ON BASIC PROGRAMMING FOR VZ USERS. SO HERE GOES.

FIRSTLY, THE SERIES WILL BE UNCONVENTIONAL. MOST INTRODUCTIONS TO BASIC PROCEED BLOW-BY-BLOW THROUGH THE VARIOUS BASIC COMMANDS. I WILL NOT — MANY TEXTS EXIST WHICH CAN EXPLAIN THESE BETTER THAN I CAN.

SECONDLY, THE SERIES WILL INITIALLY BE NON-SPECIFIC TO ANY PARTICULAR COMPUTER LANGUAGE. GENERAL PROGRAMMING CONCEPTS AND GUIDELINES WILL BE OFFERED. THE PRINCIPLES WILL BE EQUALLY APPLICABLE TO BASIC, ASSEMBLER, PASCAL OR WHATEVER.

THIRDLY, ADVANCED PROGRAMMING CONCEPTS AND HINTS WILL BE OFFERED AS THEY ARE NEEDED. THIS IS THE BEST TIME TO INTRODUCE THESE SINCE THEIR MYSTIQUE IS REMOVED.

FOURTHLY, EARLY EMPHASIS WILL BE ON PLANNING, ORGANISING AND MAINTAINING A PROGRAM, RATHER THAN ENCOURAGING FEVERISH CODING AT THE KEYBOARD (WHICH IS USUALLY COMMENCED TOO EARLY BY BEGINNERS).

IT IS QUITE POSSIBLE TO RECOGNISE A BREED OF COMPULSIVE PROGRAMMERS, BORN FROM THE HOME MICRO BOOM. THIS BREED, IS EMERGING FROM THE BRAVE NEW WORLD OF TOMORROW'S TECHNOLOGY WHOSE REASON FOR EXISTENCE IS SIMPLY TO PROGRAM. PEOPLE BECOME TOTALLY FASCINATED BY THE UNLIMITED ABSTRACT WORLD THAT THE INSIDE OF A COMPUTER OFFERS.

WE CAN CREATE A UNIVERSE OR ANY WORLD INSIDE A MACHINE. IN THE ABSTRACT WORLD OF PROGRAMMING, A WELL THOUGHT OUT PROGRAMMING METHOD SERVES AS A MAP, AND THE TECHNIQUES OF SOFTWARE ENGINEERING ARE THE WEAPONS. THESE THEN ARE THE MAIN THREADS OF THIS SERIES.

LET'S COMMENCE THIS MONTH WITH A FEW DEFINITIONS AND CONCEPTS TO PONDER OVER UNTIL THE NEXT INSTALLMENT.

THE COMPUTER IS A MACHINE, AND IS ONLY CAPABLE OF DOING SIMPLE WORK. IT HAS BEEN TERMED BY SOME AS "A REMARKABLY EFFICIENT COUNTING MACHINE WITH A LARGE MEMORY — BUT NO BRAINS!" IT HAS NO INTELLIGENCE AND CANNOT THINK.

A COMPUTER SYSTEM CONSISTS OF FOUR ELEMENTS :-

1. THE CENTRAL PROCESSOR UNIT (IN THE VZ IT IS THE Z-80A MICROPROCESSOR CHIP) WITH "PRIMARY MEMORY" (ROM AND UP TO 34K RAM).

2. INPUT DEVICES — KEYBOARD, CASSETTE, DISK AND SO ON.

3. OUTPUT DEVICES — SCREEN, PRINTER, CASSETTE, DISK, IN-BUILT SPEAKER, VOICE AND SOUND SYNTHESISERS ETC.

4. "SECONDARY MEMORY" — NOT ESSENTIAL BUT MAY BE CASSETTE OR DISK WHEN USED TO UPDATE OR RELIEVE PRIMARY MEMORY.

MAN-MACHINE INTERFACE. THE INTERACTION BETWEEN MAN-MACHINE INPUTS AND OUTPUTS IS A CONTINUOUS AND CIRCULAR FEEDBACK PROCESS. E.G. MAN OUTPUT (KEYPRESS) IS MACHINE INPUT ..OR.. MACHINE OUTPUT (SCREEN PROMPT) IS MAN INPUT- ..AND SO ON. THIS INTERACTION FORMS THE BASIS OF USING COMPUTERS.

THE FOUR FOLD SUBDIVISION OF A COMPUTER SYSTEM IS LITTLE DIFFERENT TO OUR OWN MENTAL CAPABILITIES. THE CPU AND PRIMARY MEMORY IS BROADLY EQUIVALENT TO OUR MIND. THE I/O DEVICES ARE SIMILAR TO OUR SENSES (TOUCH, TASTE, SIGHT, SENSE OF HEAT, SPEAKING, HEARING).

THE SECONDARY MEMORY IS DIRECTLY COMPARABLE TO OUR USE OF EXTERNAL AIDS TO ASSIST OUR MEMORY, SUCH AS NOTE BOOKS, FILING CABINETS OF INFORMATION, TELEPHONE DIRECTORIES - ALL OF WHICH HAVE SLOW ACCESS AND ARE DIFFICULT TO RECALL COMPARED WITH THINGS ALREADY RESIDENT IN OUR MIND.

COMPUTER PROCESSES OR CAPABILITIES ARE SURPRISINGLY FEW IN NUMBER. THERE ARE ONLY FOUR AND UNLESS AN EXERCISE OR PROBLEM CAN BE BROKEN DOWN INTO THESE ELEMENTARY PROCESSES, THEN CODING OF THE PROGRAM SHOULD NOT COMMENCE. A GREATER UNDERSTANDING OF THE PROBLEM IS REQUIRED BEFORE PROCEEDING.

IT IS IMPORTANT TO CLEARLY DISTINGUISH TWO THINGS WHILST PROGRAMMING. THE FIRST, IS TO DEVISE A LOGICAL SOLUTION TO THE PROGRAMMING EXERCISE, WHICH IS QUITE INDEPENDENT OF THE PARTICULAR LANGUAGE TO BE USED. THE SECOND, IS THE ACTUAL CODING OF THE EXERCISE BEING UNDERTAKEN. THE LATTER STAGE IS EASY, PROVIDED THAT THE FORMER IS WELL UNDERSTOOD.

THE COMPUTER PROGRAM WILL ONLY FUNCTION CORRECTLY IF THE LOGIC OF THE PROGRAM IS CORRECT, AND THERE ARE NO AIDS OR DIAGNOSTICS AVAILABLE FROM THE MACHINE TO ASSIST IN ACHIEVING CORRECTNESS IN THIS DEMANDING ASPECT OF PROGRAM DESIGN. SOME DIAGNOSTICS ARE HOWEVER AVAILABLE TO ASSIST IN THE CODING PORTION OF THE TASK - SUCH AS THE SYNTAX CHECKING.

AS ONE BECOMES MORE FAMILIAR WITH PROGRAMMING LANGUAGES IT IS SOON APPARENT THAT MANY OF THE POWERFUL COMMAND STRUCTURES ARE SIMPLY MACRO INSTRUCTIONS FORMED FROM THESE FEW "PRIMATIVES".

THE FOUR PROCESSES ARE :-

1. INPUT DATA AND STORE IT IN PRIMARY MEMORY - THE DATA MAY BE EITHER "RAW" DATA INPUT (E.G. FROM KEYBOARD) OR READ-IN FROM THE SECONDARY STORE. (E.G.TAPE).

2. OUTPUT DATA ALREADY STORED IN PRIMARY MEMORY - EITHER AS "OUTPUT" (E.G. TO SCREEN) OR WRITTEN-OUT TO SECONDARY MEMORY (E.G.TAPE).

3. PERFORM SIMPLE ARITHMETIC PROCEDURES (ADDITION OR SUBTRACTION) UPON DATA IN PRIMARY MEMORY ONLY.

4. PERFORM LOGICAL COMPARISONS (DISJUNCTION, CONJUNCTION AND NEGATION) BETWEEN TWO ITEMS OF DATA IN PRIMARY MEMORY.

(REMEMBER - I/O, ARITHMETIC, COMPARISONS ONLY)

TO CONTINUE THE ANALOGY WITH OURSELVES, I DOUBT WHETHER WE CAN DO ANYTHING MORE THAN THESE OPERATIONS EXCEPT THAT WE USE EXPERIENCE. THE COMPUTERS' ANALOGUE OF THIS IS THE PROGRAM AS IT POSSESSES ZERO INTELLIGENCE.

THE PROGRAMMING TASK IS TO UTILIZE THE HIGH SPEED AND LARGE MEMORY CAPACITY OF A COMPUTER SYSTEM TO DO SOMETHING USEFUL - SUCH AS CARRY OUT CALCULATIONS (NUMBER CRUNCHING), PLAY GAMES, MONITOR HOUSE SECURITY ETC.

THE SPECTRUM OF TASKS INVOLVED IN PROGRAMMING IS VERY BROAD, SO LITTLE WONDER THAT BEGINNERS HAVE TROUBLE GRASPING THE ESSENTIALS, OR THAT MANY PROGRAMS ARE "BADLY" WRITTEN. THE TASK INVOLVES TAKING AN IDEA OR CONCEPT AND TRANSLATING THAT INTO A SYMBOLIC (PROGRAM STATEMENT) FORM OF REPRESENTATION.

AN INTERMEDIATE STAGE IN THIS TRANSLATION OFTEN INVOLVES MODELLING THE PHENOMENON BEING PROGRAMMED. THIS PSYCHOLOGICALLY INVOLVES MOVING FROM CONCRETE CONCEPTS TO VARIOUS LEVELS OF ABSTRACTION - AGAIN A VERY DIFFICULT THING FOR, PARTICULARLY YOUNG, MINDS TO MASTER.

THE TRANSITION FROM AN IDEA TO A PROGRAM CAN SELDOM BE ACHIEVED IN ONE LEAP - MORE OFTEN A NUMBER OF INTERMEDIATE STEPS ARE REQUIRED. LIKEN IT TO WRITING AN ESSAY WHERE DRAFTS AND NOTES ARE USED BEFORE THE FINAL PROSE IS PRODUCED. FORTUNATELY A NUMBER OF USEFUL TOOLS HAVE BEEN DEVELOPED TO ASSIST IN PRODUCING A GOOD PROGRAM.

IN MY VIEW, ONE OF THE GREATEST PITFALLS OF THE HOME COMPUTER BOOM IS THAT THESE INTERMEDIATE STEPS ARE NOT UNDERSTOOD BY USERS SO THAT, AT LEAST, BAD PROGRAMS AND, AT WORST, DISILLUSIONED PROGRAMMERS RESULT. MANY OF THESE PEOPLE MAY FIND THEIR WAY INTO THE COMPUTER INDUSTRY OF THE FUTURE.
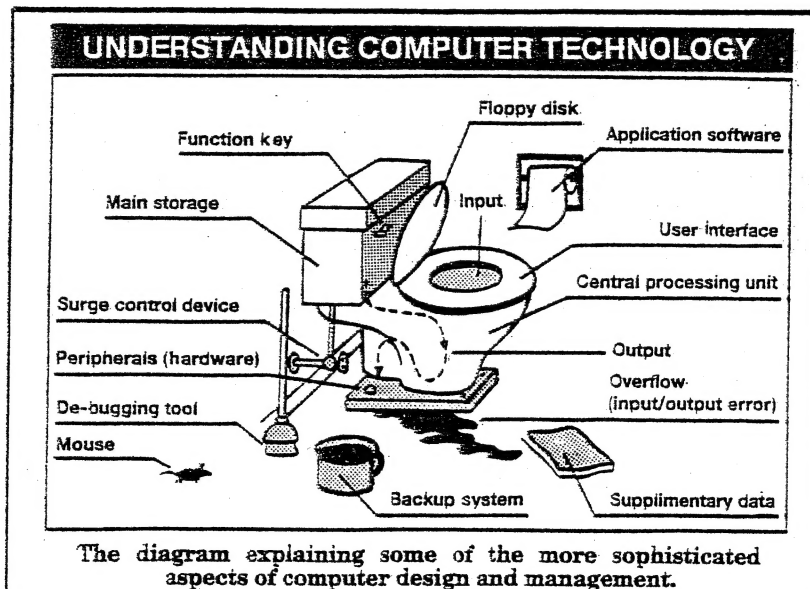
THERE IS ALWAYS MORE PERSONAL SATISFACTION IN ACHIEVING A "GOOD" JOB EVEN IF IT IS ONLY A GAMES PROGRAM FOR THE KIDS. IT IS ALSO MORE FUN, (THE ESSENCE OF HOME MICROS) AS THERE IS LESS HASSLE IN GETTING A PROGRAM TO RUN, AND MORE TIME FOR MORE PROGRAMS.

IN THE MICROCOMPUTER ENVIRONMENT WHERE THERE ARE ALWAYS HARDWARE LIMITATIONS, IT MEANS THAT IT IS VERY DIFFICULT TO COMPLETELY SEPERATE HARDWARE AND SOFTWARE ASPECTS OF THE PROGRAMMING TASK. THE PROGRAMMER MAY HAVE TO GET "CLOSE TO THE HARDWARE" - USUALLY DUE TO HARDWARE/MEMORY LIMITATIONS OR RESTRICTED I/O CAPABILITIES. DON'T SHY AWAY FROM HARDWARE BY SAYING "BUT I AM ONLY INTERESTED IN WRITING PROGRAMS" AS THE TWO ARE SOMEWHAT INSEPERABLE.

NEXT ISSUE WE WILL LOOK AT THE VARIOUS STAGES IN THE PROGRAMMING TASK, OR HOW TO APPROACH A PROGRAMMING EXERCISE. (SEE, NO MENTION OF BASIC CODE IN THIS ARTICLE!)
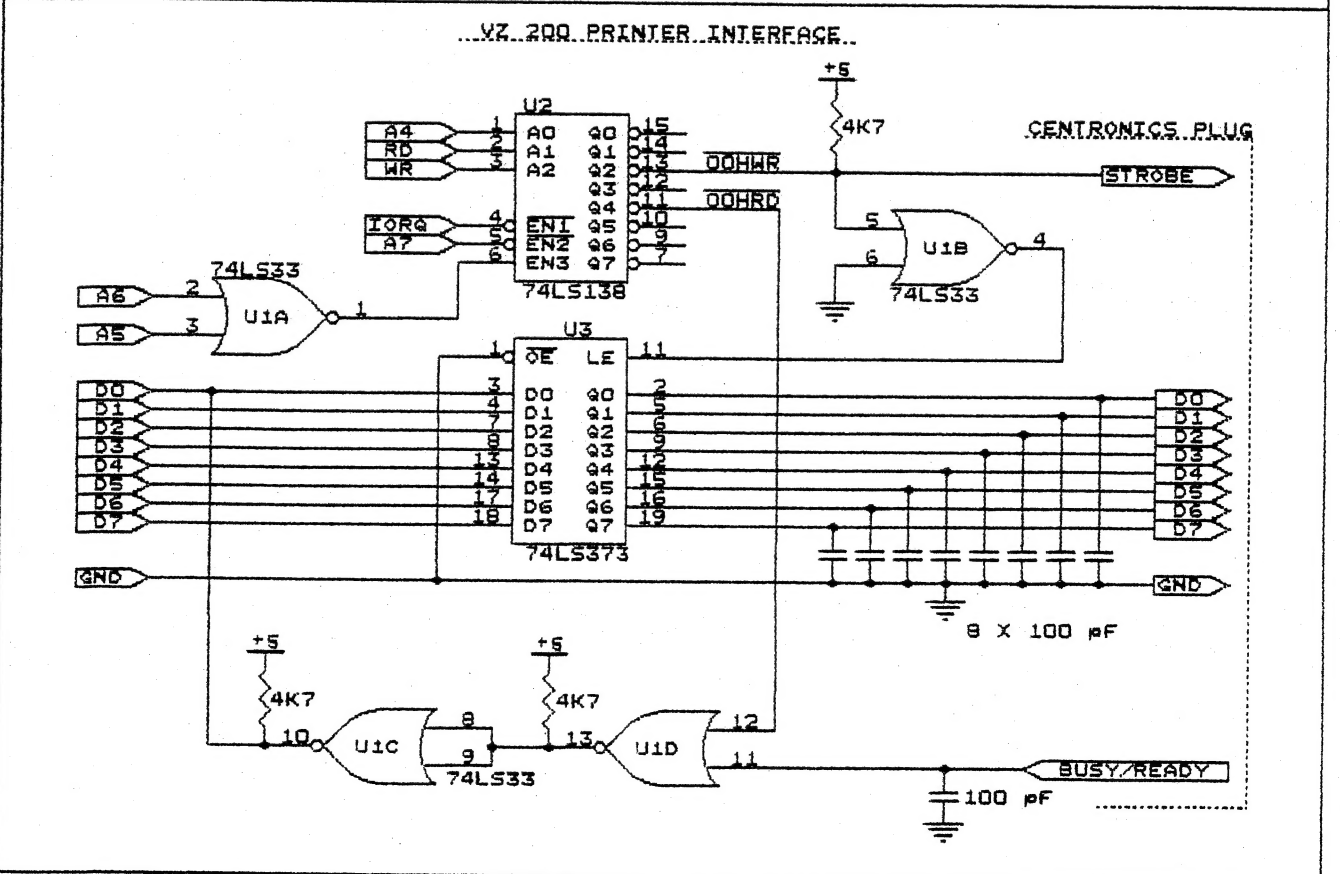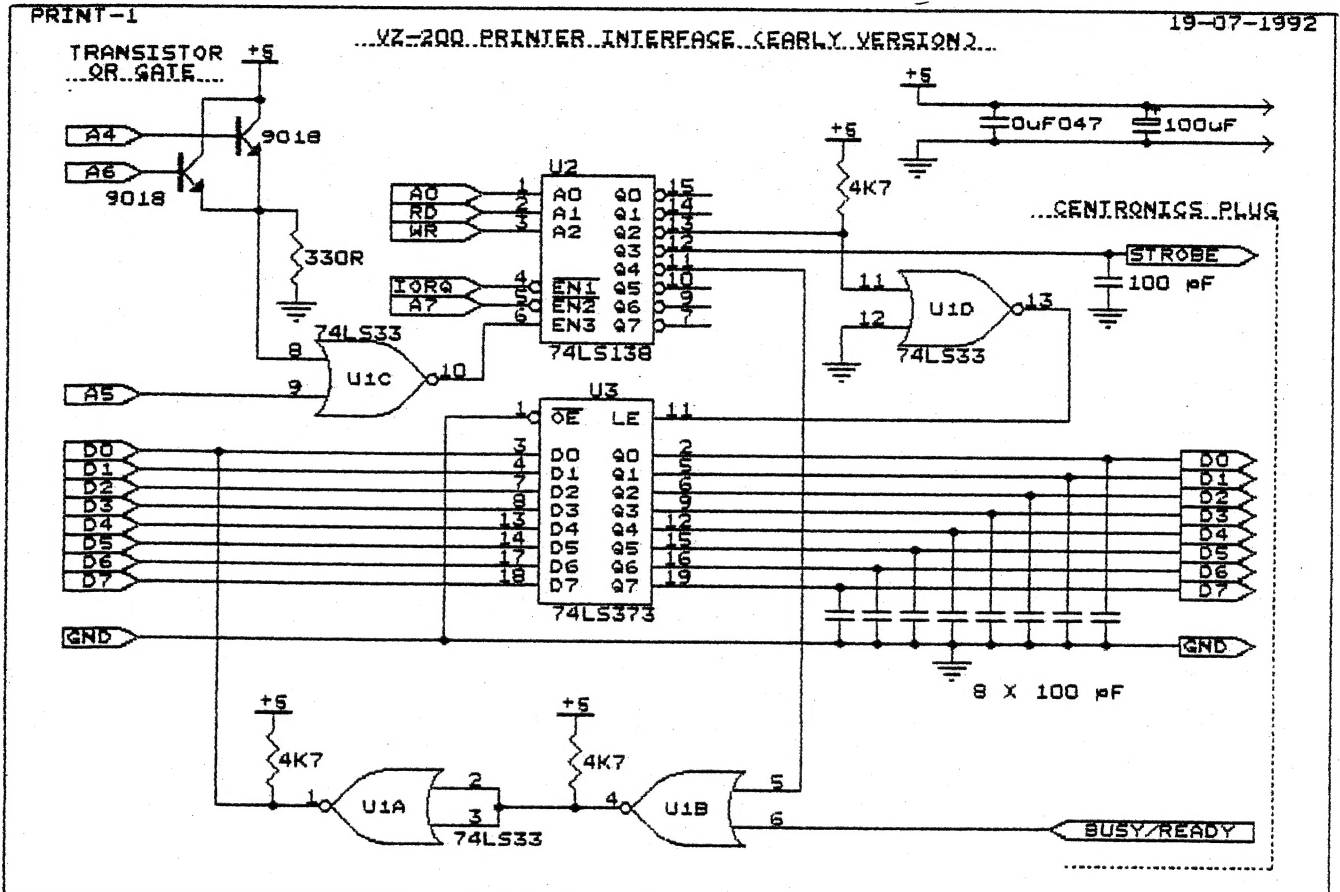
FINALLY, I WOULD LIKE TO OFFER TO USERS THAT YOUR PROGRAMMING QUERIES WILL BE ANSWERED IF YOU WRITE TO ME - WITH A SAE. PLEASE. IN THIS MANNER YOU SHOULD GET WHAT YOU WANT AND I WILL OBTAIN A FEEL FOR THE TYPE OF PROBLEMS USERS IN THE HUNTER VALLEY ARE EXPERIENCING.

WRITE TO: BOB KITCH 7 EURELLA ST KENMORE QLD 4069

UNDERSTANDING COMPUTER TECHNOLOGY

Function key
Floppy disk
Application software
Main storage
Input
User interface
Surge control device
Central processing unit
Peripherals (hardware)
Output
De-bugging tool
Overflow (input/output error)
Mouse
Backup system
Supplimentary data

The diagram explaining some of the more sophisticated aspects of computer design and management.

VZ200 PRINTER INTERFACE
(EARLY VERSION)

VZ200 PRINTER INTERFACE

PRINT-1   19-07-1992

VZ-200 PRINTER INTERFACE (EARLY VERSION)

VZ 200 PRINTER INTERFACE

Schematic diagrams showing memory address decoding circuits:
- 64K DECODING IN 2K BLOCKS (74LS138, 74LS04)
- 64K DECODING IN 4K BLOCKS (74LS138, 74LS04)
- 64K DECODING IN 8K BLOCKS (74LS138)
- 64K DECODING IN 16K BLOCKS (74LS139, U1A)
- 64K DECODING IN 32K BLOCKS (74LS139, U1B)

DECODE-4
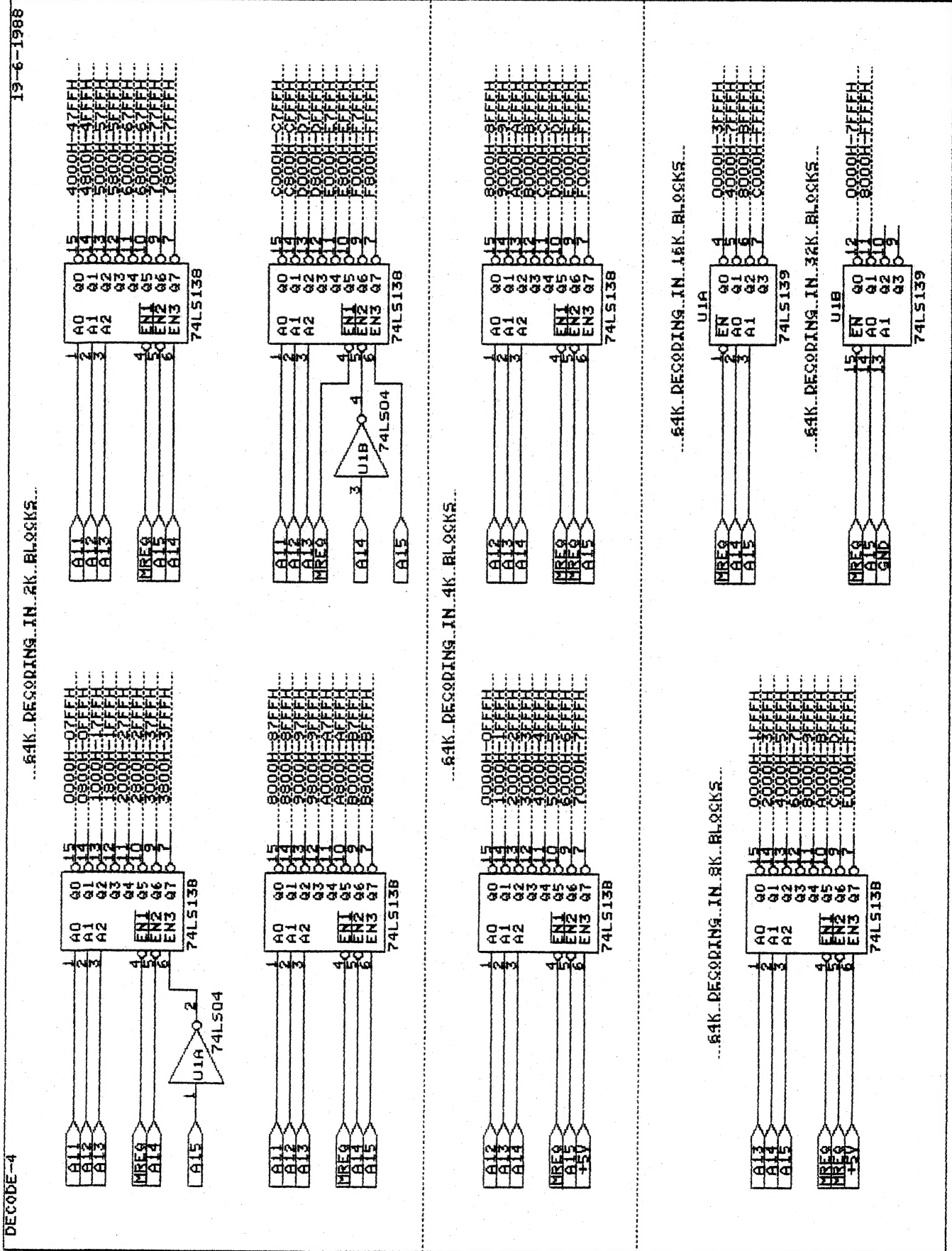
E & F WP PATCH 3.3: $20.00 PATCH 3.3 WRITTEN BY DAVE MITCHELL WILL CONVERT YOUR E & F TAPE WORD PROCESSOR FOR FULL DISK USE WHILE RETAINING ALL ORIGINAL FUNCTIONS. IT ALSO HAS SHIFT LOCK AND PRINTER CONTROL CODES WHICH CAN BE IMBEDDED IN TEXT AND SAVED TO TAPE OR DISK. **BSTWP.F:** THIS UTILITY PROVIDED WITH PATCH 3.3 WILL CONVERT BASIC PROGRAMS AND ED/ASS. SOURCE CODE FILES INTO WORD PROCESSOR FILES.

EXTENDED DOS V1.3: $15.00 THESE COMMANDS ARE AT YOUR DISPOSAL: MERGE, DIRA, DIRA, DIRB, LDIRB, OLD, OLD., DEC, HEX, MENU, CODE, LTAB, MOVE AND UPDATE, STATUSA AND LSTATUSA. STATUSA AND LSTATUSA ALSO WORKS WITH VERSION 1.0 DOS

MENU/FILE COPIER - $15.00 THIS UTILITY WILL READ YOUR DISK DIRECTORY AND PRESENT YOU WITH SEVERAL OPTIONS. USING THE CURSOR YOU CAN RUN/BRUN ANY PROGRAM OR SELECT FILE COPY, REN, ERASE, DRIVE 1 OR 2, ETC. BESIDES COPYING TEXT AND BINARY FILES ALL OTHER FILES CAN BE COPIED AS WELL EXEPT FOR DATA FILES.

PRICES INCLUDE POSTAGE - FOR PURCHASE OR INFORMATION CONTACT:
DAVE MITCHELL 24 ELPHINSTONE STREET NORTH ROCKHAMPTON 4701
QUEENSLAND AUSTRALIA - PHONE: (079) 27 8519

CONTRIBUTIONS TO THE JOURNAL

IF YOU ARE THINKING OF CONTRIBUTING TO THE JOURNAL THE PREFERED FORMAT IS BASIC LISTINGS, WORD PROCESSOR OR SOURCE CODE FILES ON TAPE OR DISK. FILES FROM THE FOLLOWING WORD PROCESSORS CAN BE ACCEPTED :-

E & F TAPE OR DISK PATCH 3.1-3.3, WORDPRO CARTRIDGE, WORDPRO PATCH, ALL SOURCE CODE FILES AND ALL QUICKWRITE WORD PROCESSOR FILES.

CLUB MEETINGS - ALL WELCOME
FIRST FRIDAY OF MONTH

MEETINGS WILL BE ONCE A MONTH. BECAUSE SOME LOCAL MEMBERS WORK SHIFTWORK MEETING DATES WILL BE ADJUSTED TO ACCOMODATE THEM. WHETHER YOU ARE A LOCAL MEMBER, INTRA OR INTERSTATE VISITOR PLEASE CHECK WITH JOE LEON FIRST.

HUNTER VALLEY VZ JOURNAL SUBSCRIPTIONS

SUBSCRIPTION TO - AUST. - 3 ISSUES $11.00 - 6 ISSUES $21.00
H.V.VZ.JOURNAL - N. Z. - 3 ISSUES $13.00 - 6 ISSUES $26.00
PRICES INCLUDE POST & PACKING

FOR MORE INFORMATION ON MEETINGS AND SUBS CONTACT:
JOE LEON 33 TIGHES TCE TIGHES HILL 2297 (049) 692 399 AUSTRALIA

VZ USER GROUPS & PUBLICATIONS

VZ DOWN UNDER - ISSUE # 40 WILL BE THE LAST
NOTE: SEE PAGE 3 FOR MORE DETAILS.

WAVZ - WESTERN AUSTRALIA VZ USER GROUP
GRAEME BYWATER   PO BOX 388   MORLEY W A 6062

BRISBANE VZ USERS WORKSHOP - C/O 63 TINGALPA ST. WYNUM WEST 4178
SOFTWARE FOR SALE - DISK MENU

SAPPHIRE PRODUCTIONS - VZ DISK MAGAZINE - PUBLIC DOMAIN
NOTE: VZ DISK MAGAZINE HAS CEASED PRODUCTION

NOTE: WHEN WRITING TO ANY ABOVE OR H.V.VZ. USERS' GROUP FOR INFORMATION PLEASE ENCLOSE A S.S.A.E. OR NZ 2 INT. REPLY COUPONS.